

## CLAIMS

1           1.       A method for transferring an at least partly compressible file from a server  
2 computer to a user computer, the server computer and user computer being connected  
3 via a communication network, the method comprising the steps of:

4                   at least partly compressing the at least partly compressible file to produce  
5 a corresponding compressed file;

6                   storing the compressed file on the server computer;

7                   upon request, transferring the compressed file to the user computer; and

8                   decompressing the compressed file at the user computer to reconstruct  
9 the at least partly compressible file.

1           2.       A method according to claim 1 and further comprising, after the step of  
2 decompressing, the step of converting the reconstructed file into a form presentable to  
3 the user.

1           3.       A method according to claim 1 wherein the at least partly compressible file  
2 comprises a streamed-media file.

1           4.       A method according to claim 3 wherein the streamed-media file comprises  
2 a Macromedia Flash™ file.

1 5. A method according to claim 3 wherein the streamed-media file comprises  
2 a Macromedia Shockwave™ file.

1

1 6. A method according to claim 3 wherein the streamed-media file comprises  
2 a Java Applet™ file.

1

1 7. A method according to claim 1 wherein the step of compressing said at  
2 least partly compressible file comprises the steps of:

3 applying at least one predetermined type of compression algorithm to a  
4 data chunk of a predetermined size in said at least partly compressible file;

5 if the compression ratio obtained from applying said at least one type of  
6 algorithm exceeds a predetermined threshold, writing the compressed data chunk to a  
7 memory;

8 else, writing the data chunk uncompressed to said memory; and

9 repeating the above steps for a plurality data chunks in said at least partly  
10 compressible file.

1

1 8. A method according to claim 7 wherein the step of applying at least one  
2 type of compression algorithm to the at least partly compressible file comprises the  
3 steps of:

4 applying a first compression algorithm or combination of compression  
5 algorithms to said data chunk;

6 applying a second compression algorithm or combination of compression  
7 algorithms to said data chunk;  
8 determining which of said first and second algorithms or combinations of  
9 algorithms yields a higher compression ration for said chunk;  
10 using the higher compression ratio algorithm or combination of algorithms  
11 to compress said chunk.

1

1 9. A method according to claim 8 wherein at least one of said first and  
2 second compression algorithms or combinations of compression algorithms comprises  
3 a dictionary table type compression algorithm.

1

1 10. A method according to claim 8 wherein at least one of said first and  
2 second compression algorithms or combinations of compression algorithms comprises  
3 a Huffman tree type compression algorithm.

1

1 11. A method according to claim 1 wherein the step of decompressing said  
2 compressed file comprises the steps of:

3 collecting a chunk of data of said compressed file, said chunk having a  
4 predetermined size;

5 determining whether said data chunk has been compressed relative to the  
6 at least partly compressible file;

7

8 if the said data chunk is determined not to have been compressed, writing  
9 said data chunk to a memory;  
10 else, decompressing said data chunk to produce a corresponding  
11 decompressed data chunk and writing the decompressed data chunk to said memory;  
12 and  
13 repeating the above steps for a plurality data chunks of said compressed  
14 file.

1  
1 12. A method according to claim 11, wherein the step of decompressing said  
2 data chunk comprises the step of applying at least one type of decompression algorithm  
3 to the compressed data chunk.

1  
1 13. A method according to claim 12 wherein the step of applying at least one  
2 type of decompression algorithm to the at least partly compressible file comprises the  
3 steps of:  
4 determining at least one type of compression algorithm that has been used  
5 to compress said data chunk; and  
6 selecting said at least one decompression algorithm to match said at least  
7 one compression algorithm.

1  
1 14. A method for preparing an at least partly compressible file for accelerated  
2 transfer from a server computer to a user computer, the server computer and user

3 computer being connected via a communication network, the method comprising the  
4 steps of:

5 at least partly compressing the at least partly compressible file to produce  
6 a corresponding compressed file; and  
7 storing the compressed file on the server computer.

1

1 15. A method according to claim 14 wherein the at least partly compressible  
2 file comprises a streamed-media file.

1

1 16. A method according to claim 15 wherein the streamed-media file  
2 comprises a Macromedia Flash™ file.

1

1 17. A method according to claim 15 wherein the streamed-media file  
2 comprises a Macromedia Shockwave™ file.

1

1 18. A method according to claim 15 wherein the streamed-media file  
2 comprises a Java Applet™ file.

1

1 19. A method according to claim 14 wherein the step of compressing said at  
2 least partly compressible file comprises the steps of:

3 applying at least one predetermined type of compression algorithm to a  
4 data chunk of a predetermined size in said at least partly compressible file;

5 if the compression ratio obtained from applying said at least one type of  
6 algorithm exceeds a predetermined threshold, writing the compressed data chunk to a  
7 memory;  
8 else, writing the data chunk uncompressed to said memory; and  
9 repeating the above steps for a plurality data chunks in said at least partly  
10 compressible file.

1

1 20. A method according to claim 19 wherein the step of applying at least one  
2 type of compression algorithm to the at least partly compressible file comprises the  
3 steps of:

4 applying a first compression algorithm or combination of compression  
5 algorithms to said data chunk;

6 applying a second compression algorithm or combination of compression  
7 algorithms to said data chunk;

8 determining which of said first and second algorithms or combinations of  
9 algorithms yields a higher compression ration for said chunk;

10 using the higher compression ratio algorithm or combination of algorithms  
11 to compress said chunk.

1

1 21. A method according to claim 20 wherein at least one of said first and  
2 second compression algorithms or combinations of compression algorithms comprises  
3 a dictionary table type compression algorithm.

1

1           22.    A method according to claim 20 wherein at least one of said first and  
2 second compression algorithms or combinations of compression algorithms comprises  
3 a Huffman tree type compression algorithm.

1

1           23.    A method of downloading a compressed file corresponding to a  
2 compressible file in a given format from a server computer to a user computer, the  
3 server computer and user computer being connected via a communication network, the  
4 method comprising the steps of:

5                   receiving a request for the at least partly compressible file;

6                   replacing the request for the at least partly compressible file with a request  
7 for the corresponding compressed file;

8                   transferring the compressed file to the user computer; and

9                   decompressing the compressed file at the user computer to reconstruct  
10 the at least partly compressible file.

1

1           24.    A method according to claim 23 and further comprising, after the step of  
2 decompressing, the step of converting the reconstructed file into a form presentable to  
3 the user.

1

1           25.    A method according to claim 23 wherein the at least partly compressible  
2 file comprises a streamed-media file.

1  
1  
2  
1  
1  
2  
1  
1  
2  
1  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12

26. A method according to claim 25 wherein the streamed-media file comprises a Macromedia Flash™ file.

27. A method according to claim 25 wherein the streamed-media file comprises a Macromedia Shockwave™ file.

28. A method according to claim 25 wherein the streamed-media file comprises a Java Applet™ file.

29. A method according to claim 23 wherein the step of decompressing said compressed file comprises the steps of:

collecting a chunk of data of said compressed file, said chunk having a predetermined size;

determining whether said data chunk has been compressed relative to the at least partly compressible file;

if the said data chunk is determined not to have been compressed, writing said data chunk to a memory;

else, decompressing said data chunk to produce a corresponding decompressed data chunk and writing the decompressed data chunk to said memory;  
and



13 repeating the above steps for a plurality data chunks of said compressed  
14 file.

15 30. A method according to claim 29, wherein the step of decompressing said  
16 data chunk comprises the step of applying at least one type of decompression algorithm  
17 to the compressed data chunk

1

1 31. A method according to claim 30 wherein the step of applying at least one  
2 type of decompression algorithm to the at least partly compressible file comprises the  
3 steps of:

4 determining a type of compression algorithm or combination of  
5 compression algorithms that have been used to compress said data chunk; and

6 selecting said at least one decompression algorithm to match said  
7 compression algorithm or combination of compression algorithms.

1

1 32. A system for transferring an at least partly compressible file from a server  
2 computer to a user computer, the server computer and user computer being connected  
3 via a communication network, the system comprising:

4 a compress program which at least partly compresses the at least partly  
5 compressible file to produce a corresponding compressed file;

6 a memory associated with the server computer for storing said  
7 compressed file;

8 means associated with the user computer for receiving a request for said

9 compressible file and transmitting a request for said corresponding compressed file from  
10 the server computer via said communication link; and

11 a decompress program, associated with the user computer, which  
12 decompresses the compressed file to reconstruct the at least partly compressible file.

1

1 33. A system according to claim 32 and further comprising means for  
2 converting the reconstructed compressible file into a form presentable to the user.

1

1 34. A system according to claim 33 wherein the at least partly compressible  
2 file comprises a streamed-media file.

1

1 35. A system according to claim 33 wherein the streamed-media file comprises  
2 a Macromedia Flash™ file.

1

1 36. A system according to claim 33 wherein the streamed-media file comprises  
2 a Macromedia Shockwave™ file.

1

1 37. A system according to claim 33 wherein the streamed-media file comprises  
2 a Java Applet™ file.

1

1 38. A system according to claim 32 wherein said compress program  
2 comprises:

3 means for applying at least one predetermined type of compression  
4 algorithm to each of a plurality of data chunks of a predetermined size in said at least  
5 partly compressible file to produce corresponding compressed data chunk;

6 means for writing the compressed data chunk to said memory of the server  
7 computer if the compression ratio obtained from applying said at least one type of  
8 algorithm exceeds a predetermined threshold; and

9 means for writing the data chunk uncompressed to said memory of the  
10 server computer if the compression ratio obtained from applying said at least one type  
11 of algorithm does not exceed said predetermined threshold.

1

1 39. A system according to claim 38 wherein said means of applying at least  
2 one type of compression algorithm to each of the a plurality of chunks of at least partly  
3 compressible file comprises:

4 means for applying a first compression algorithm or combination of  
5 compression algorithms to said data chunk;

6 means for applying a second compression algorithm or combination of  
7 compression algorithms to said data chunk; and

8 means for determining which of said first and second algorithms or  
9 combinations of algorithms yields a higher compression ration for said chunk,

10 wherein the higher compression ratio algorithm or combination of  
11 algorithms is used to compress said data chunk.

1

1 40. A system according to claim 39 wherein at least one of said first and  
2 second compression algorithms or combinations of compression algorithms comprises  
3 a dictionary table type compression algorithm.

1

1 41. A system according to claim 39 wherein at least one of said first and  
2 second compression algorithms or combinations of compression algorithms comprises  
3 a Huffman tree type compression algorithm.

1

1 42. A system according to claim 32 wherein said decompress program  
2 comprises:

3 means for collecting a plurality of data chunks of said compressed file, said  
4 data chunks having a predetermined size;

5 means for determining whether each of said data chunks has been  
6 compressed relative to the at least partly compressible file;

7 means for writing each data chunk which is determined not to have been  
8 compressed to a memory of the user station;

9 means for decompressing each data chunk which is determined to have  
10 been compressed to produce a corresponding decompressed data chunk and writing  
11 each decompressed data chunks to said memory.

1

1 43. A method according to claim 42, wherein said decompress program  
2 comprises means for applying at least one type of decompression algorithm to the

3 compressed data chunk.

1

1 44. A method according to claim 43 wherein said means for applying at least  
2 one type of decompression algorithm to the at least partly compressible file comprises:  
3 means for determining at least one type of compression algorithm that has  
4 been used to compress said data chunk; and  
5 means for selecting said at least one decompression algorithm to match  
6 said at least one compression algorithm.

1

2

00577-1656260